Remarks

The present amendment responds to the Official Action mailed June 4, 2003. That action objected to the specification and the drawings as being informal. Claims 6 and 8 were objected to due to informalities. Claims 1, 2, 6-8 and 11 were rejected under 35 U.S.C. 102(b) as being anticipated by Lee et al. U.S. Patent No. 4,763,242 ("Lee"). Claims 3 and 4 were rejected under 35 U.S.C. 103(a) as being unpatentable over Lee in view of Dowling U. S. Patent No. 6,128,728 ("Dowling '728"). Claim 5 was rejected under 35 U.S.C. 103(a) as being unpatentable over Lee in view of Dowling U.S. Patent No. 6,170,051 ("Dowling '051"). Claims 8-11 were rejected under 35 U.S.C. 103(a) as being unpatentable over Dahl et al. U.S. Patent No. 5,710,938 ("Dahl") in view of Lee. Claims 12 and 13 were rejected under 35 U.S.C. 103(a) as being unpatentable over Dahl in view of Lee, in further view of Mirsky et al. U.S. Patent No. 5,915,123 ("Mirsky").

Claims 1, 6 and 8-12 have been amended to be more clear and distinct. Claims 1-13 are presently pending.

The Objections to the Specification

The specification has been updated to include the serial numbers of the referenced applications.

At page 8, line 12, "PE/PEs" has been replaced with "PEs" to correct a typographical error.

At page 8, line 19, "a" has been replaced with "an" to correct a typographical error.

On page 9, the paragraph beginning at line 16 has been deleted.

During the preparation of formal drawings, Fig. 5 was divided into Figs. 5A and 5B. The specification has been amended to reflect this change.

The Official Action indicated that the text at page 11, lines 6-8, is unclear to the Examiner. As shown in Fig. 3, a 1x1 core processor can be configured to operate as a 1x1 or a 1x0 core processor depending on the CSB bit. The cited text simply clarifies that while the core processor can be configured in 1x1 or 1x0 mode, when no PE instructions are used in the program, the core processor is effectively operating as a 1x0 irrespective of whether the actual configuration is 1x1 or 1x0. If the Examiner wishes to discuss this point further, he is encouraged to call the undersigned at his convenience.

The Objections to the Drawings

Figs. 1 and 4 have been amended to correct typographical errors.

Regarding the Examiner's question as to whether the term "C-bit instruction bus" of Figs. 1, 4 and 5A should be changed to "B-bit instruction bus" to match the memory size of B-bits, the term "C-bit" is used to indicate that the instruction bus does not necessarily match the memory size. For example, with support for instruction abbreviation, instruction bus width may be less than the memory width.

The Claim Objections

Claims 6 and 8 have been amended to be more clear and distinct.

The Art Rejections

These rejections are respectfully traversed as not supported by the relied upon art. The relied upon art does not anticipate and does not render obvious the claims as presently amended, as addressed in greater detail below.

Lee describes a system which includes a main processor and an assist, or hardware which extends the main processor's capabilities by providing support for additional extension instructions which are not part of the main processor's basic instruction set. This assist may be in

the form of a coprocessor or a special function unit. When an instruction is fetched from memory, a field in the instruction is decoded to determine whether the instruction is a basic instruction or an extension instruction. If the instruction is a basic instruction, it is executed by the main processor. If the instruction is an extension instruction, the field is further decoded to determine which assist to route the instruction to for execution.

In contrast to Lee, the present provides techniques for efficient context switching between tasks executing in an array processing environment. In one aspect of the present invention, an array controller sequence processor (SP) is merged with a processing element (PE) in order to share execution units between the SP and PE. Consequently, in the merged SP/PE a single set of execution units are coupled with two independent register files. To make efficient use of the SP and PE resources, a bit in the instruction format, the SP/PE bit, differentiates SP instructions from PE instructions. Multiple register contexts may be obtained by controlling how the SP/PE bit in the instruction format is used in conjunction with a context switch bit (CSB) for the context selection of the PE register file or the SP register file. See amended claim 1, for example, which recites " a first set of registers stored in a first register file; a second set of registers stored in a second register file; a sequence processor/processing element (SP/PE) selection bit in an instruction; and *a context select bit (CSB) in a processor state register selecting a context of a first software task or a context of a second software tas*k which in conjunction with the SP/PE selection bit determines which set of registers is to be accessed by the instruction." (emphasis added) See also amended claims 8 and 11, for example. Lee does not teach and does not render obvious such a technique for context switching. The assist bit field imbedded in the instruction of Lee indicates which hardware assist should be used for executing the instruction, and is not a context select bit stored in a processor state register, as presently claimed.

Regarding claims 8-11, Dahl does not cure the failings of Lee as a reference. Dahl teaches an array system in which the array is partitioned into multiple sub-arrays which operate independently of each other. As indicated by the Official Action, Dahl does not teach apparatus for providing efficient context switching between tasks. Applicants agree. As addressed above, Lee also fails to teach context switching between tasks as presently claimed. Thus, the proposed combination of these two items does not teach and does not make obvious the present claims.

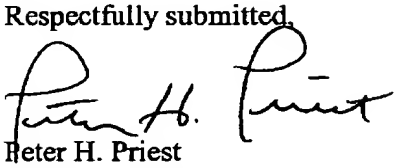
Regarding dependent claims 3-5, 12 and 13, the other relied upon references do not cure the failings of Lee as a reference. Dowling '728 teaches a system which utilizes shadow register sets and shadow windows to transfer data between registers and memory in burst and cycle steal modes. Dowling '051 describes a processor which allows one program to execute in a cycle steal to make use of the inefficiencies of another program. Mirsky teaches techniques for providing local control of processing elements in a network of multiple context processing elements. These references do not teach and do not render obvious the presently claimed techniques for context switching.

In summary, the relied upon art does not indicate a recognition of the problems addressed by the present invention. Further, the relied upon art does not teach or suggest an apparatus which would solve the problems of context switching on array processors addressed by the present invention in the manner solved by the present invention. The claims as presently amended are not taught, are not inherent, and are not obvious in light of the relied upon art.

Conclusion

Any questions regarding this application may be raised by telephone with the undersigned if it is considered that processing of this application will be expedited thereby.

Respectfully submitted,



Peter H. Priest
Reg. No. 30,210
Priest & Goldstein, PLLC
5015 Southpark Drive, Suite 230
Durham, NC 27713
(919) 806-1600